In the claims:

1 - 3. (Canceled)

4. (Currently amended) A method of operating a cryptographic data processing system that comprises a host processor, a system memory coupled to the host processor, and a cryptographic processor integrated circuit that comprises a local memory and is coupled to the host processor and the system memory, the method comprising:

providing a command queue in the system memory;

loading a command block into the command queue using the host processor;

executing the command block using the cryptographic processor; and

notifying the host processor that the command block has been executed by updating a completion field in the command block using the cryptographic processor.

5. (Original) A method as recited in Claim 4, further comprising:

providing a read address for the command queue and a write address for the command queue;

wherein loading the command block into the command queue using the host processor comprises loading the command block into the command queue using the host processor beginning at the write address, and wherein executing the command block using the cryptographic processor comprises executing the command block using the cryptographic processor beginning at the read address.

6. (Original) A method as recited in Claim 5, wherein loading the command block into the command queue using the host processor beginning at the write address comprises:

determining if the write address plus an amount corresponding to a size of a single command block equals the read address; and

loading the command block into the command queue using the host processor beginning at the write address if the write address plus the amount corresponding to the size of the single command block does not equal the read address.

7. (Original) A method as recited in Claim 6, further comprising:

incrementing the write address by the amount corresponding to the size of a single command block using the host processor after loading the command block into the command queue using the host processor beginning at the write address if the write address plus the amount corresponding to the size of the single command block does not equal the read address.

8. (Original) A method as recited in Claim 5, wherein executing the command block using the cryptographic processor beginning at the read address comprises:

determining whether the read address is equal to the write address; and

executing the command block using the cryptographic processor beginning at the read address if the read address is not equal to the write address.

9. (Original) A method as recited in Claim 8, further comprising:

incrementing the read address by an amount corresponding to a size of a single command block using the cryptographic processor after executing the command block using the cryptographic processor beginning at the read address.

10. (Original) A method as recited in Claim 4, wherein notifying the host processor that the command block has been executed comprises invoking an interrupt using the cryptographic processor after executing the command block.

11. (Canceled)

12. (Currently amended) A method as recited in Claim ~~11~~ 4, further comprising:

providing a periodic interrupt; and

reading the completion field using the host processor upon invocation of the periodic interrupt.

13. (Original) A method as recited in Claim 4, wherein notifying the host processor that the command block has been executed comprises:

setting a timer after loading the command block into the command queue using the host processor; and

checking whether the command block has been executed after expiration of the timer.

14. (Original) A method as recited in Claim 4, further comprising:

loading at least one operand from the command queue to the local memory;

performing at least one operation on the at least one operand to generate a result in the local memory; and

storing the result generated in the local memory in the command queue.

15. (Original) A method of operating a cryptographic data processing system that comprises a host processor, a system memory coupled to the host processor, and a cryptographic processor integrated circuit that is coupled to the host processor and the system memory, the method comprising:

providing a command queue in the system memory;

loading a command block into the command queue using the host processor;

setting a value of an interrupt field in the command block to request an interrupt when the command block has been executed;

executing the command block using the cryptographic processor; and

invoking an interrupt using the cryptographic processor after executing the command block if the interrupt field in the command block is set to the value to request the interrupt.

16. (Original) A method as recited in Claim 15, further comprising:

storing error information in the command block that is associated with executing the command block using the cryptographic processor.

17 - 19. (Canceled)

20. (Original) A method of operating a data processing system that comprises a host processor, a system memory coupled to the host processor, and an adjunct processor integrated circuit that is coupled to the host processor and the system memory, the method comprising:

providing a command queue in the system memory;

loading a command block into the command queue using the host processor, the command block comprising an input data field that contains input data;

performing an operation based on the input data using the adjunct processor to generate a result; and

storing the result in the input data field such that at least a portion of the input data is overwritten.

21. (Original) A method as recited in Claim 20, wherein the data processing system comprises a cryptographic data processing system, the adjunct processor integrated circuit comprises a cryptographic processor integrated circuit, and performing the operation based on the input data comprises:

performing a hash operation based on the input data using the cryptographic processor to generate a hash value.

22. (Original) A method as recited in Claim 21, wherein storing the result in the input data field comprises:

storing the hash value in the input data field such that the at least a portion of the input data is overwritten.

23. (Original) A method as recited in Claim 21, wherein the command block further comprises an input pointer field that contains an address in the system memory of an incoming packet and wherein performing the hash operation comprises:

performing the hash operation based on the input data and the incoming packet using the cryptographic processor to generate the hash value.

24. (Original) A method as recited in Claim 23, wherein the command block further comprises an output pointer field that contains an address in the system memory for storing a decrypted packet, the method further comprising:

decrypting the incoming packet using the cryptographic processor to generate the decrypted packet;

attaching the hash value to the decrypted packet; and

storing the decrypted packet with the attached hash value at the address in the system memory contained in the output pointer field.

25. (Original) A method of operating a cryptographic data processing system that comprises a host processor, a system memory coupled to the host processor, and a cryptographic processor integrated circuit that comprises a local memory and is coupled to the host processor and the system memory, the method comprising:

providing a command queue in the system memory;

providing a read address for the command queue and a write address for the command queue;

loading a random number sample into the command queue using the cryptographic processor beginning at the write address; and

reading the random number sample using the host processor beginning at the read address.

26. (Original) A method as recited in Claim 25, wherein loading the random number sample into the command queue using the cryptographic processor beginning at the write address comprises:

determining if the write address plus an amount corresponding to a size of a single random number sample equals the read address; and

loading the random number sample into the command queue using the cryptographic processor beginning at the write address if the write address plus the amount corresponding to the size of the single random number sample does not equal the read address.

27. (Original) A method as recited in Claim 26, further comprising:

incrementing the write address by the amount corresponding to the size of a single random number sample using the cryptographic processor after loading the random number sample into the command queue using the cryptographic processor beginning at the write address if the write address plus the amount corresponding to the size of the single random number sample does not equal the read address.

28. A method as recited in Claim 25, wherein reading the random number sample using the host processor beginning at the read address comprises:

determining whether the read address is equal to the write address; and

reading the random number sample using the host processor beginning at the read address if the read address is not equal to the write address.

29. (Original) A method as recited in Claim 28, further comprising:

incrementing the read address by an amount corresponding to a size of a single random number sample using the host processor after reading the random number sample using the host processor beginning at the read address.

30 - 32. (Canceled)

33. (Currently Amended) A cryptographic data processing system that comprises a host processor, a system memory coupled to the host processor, and a cryptographic processor integrated circuit that comprises a local memory and is coupled to the host processor and the system memory, the system further comprising:

means for providing a command queue in the system memory;

means for loading a command block into the command queue using the host processor;

means for executing the command block using the cryptographic processor; and

means for notifying the host processor that the command block has been executed, the means for notifying comprising means for updating a completion field in the command block using the cryptographic processor.

34. (Original) A cryptographic data processing system as recited in Claim 33, further comprising:

means for providing a read address for the command queue and a write address for the command queue;

wherein the means for loading the command block into the command queue using the host processor comprises means for loading the command block into the command queue using the host processor beginning at the write address, and wherein the means for executing the command block using the cryptographic processor comprises means for executing the command block using the cryptographic processor beginning at the read address.

35. (Original) A cryptographic data processing system as recited in Claim 34, wherein the means for loading the command block into the command queue using the host processor beginning at the write address comprises:

means for determining if the write address plus an amount corresponding to a size of a single command block equals the read address; and

means for loading the command block into the command queue using the host processor beginning at the write address if the write address plus the amount corresponding to the size of the single command block does not equal the read address.

36. (Original) A cryptographic data processing system as recited in Claim 35, further comprising:

means for incrementing the write address by the amount corresponding to the size of a single command block using the host processor if the write address plus the amount corresponding to the size of the single command block does not equal the read address, the means for incrementing being responsive to the means for loading the command block into the command queue using the host processor beginning at the write address.

37. (Original) A cryptographic data processing system as recited in Claim 34, wherein the means for executing the command block using the cryptographic processor beginning at the read address comprises:

means for determining whether the read address is equal to the write address; and means for executing the command block using the cryptographic processor beginning at the read address if the read address is not equal to the write address.

38. (Original) A cryptographic data processing system as recited in Claim 37, further comprising:

means for incrementing the read address by an amount corresponding to a size of a single command block using the cryptographic processor, the means for incrementing being responsive to the means for executing the command block using the cryptographic processor beginning at the read address.

39. (Original) A cryptographic data processing system as recited in Claim 33, wherein the means for notifying the host processor that the command block has been executed comprises means for invoking an interrupt using the cryptographic processor after executing the command block.

40. (Canceled)

41. (Currently amended) A cryptographic data processing system as recited in Claim ~~40~~ 33, further comprising:

means for providing a periodic interrupt; and

means for reading the completion field using the host processor upon invocation of the periodic interrupt.

42. (Original) A cryptographic data processing system as recited in Claim 33, wherein the means for notifying the host processor that the command block has been executed comprises:

means for setting a timer after loading the command block into the command queue using the host processor; and

means for checking whether the command block has been executed after expiration of the timer.

43. (Original) A cryptographic data processing system as recited in Claim 33, further comprising:

means for loading at least one operand from the command queue to the local memory;

means for performing at least one operation on the at least one operand to generate a result in the local memory; and

means for storing the result generated in the local memory in the command queue.

44. (Original) A cryptographic data processing system that comprises a host processor, a system memory coupled to the host processor, and a cryptographic processor integrated circuit that is coupled to the host processor and the system memory, the system further comprising:

means for providing a command queue in the system memory;

means for loading a command block into the command queue using the host processor;

means for setting a value of an interrupt field in the command block to request an interrupt when the command block has been executed;

means for executing the command block using the cryptographic processor; and

means for invoking an interrupt using the cryptographic processor after executing the command block if the interrupt field in the command block is set to the value to request the interrupt.

45. (Original) A cryptographic data processing system as recited in Claim 44, further comprising:

means for storing error information in the command block that is associated with executing the command block using the cryptographic processor.

46 - 48. (Canceled)

49. (Original) A data processing system that comprises a host processor, a system memory coupled to the host processor, and an adjunct processor integrated circuit that is coupled to the host processor and the system memory, the system further comprising:

means for providing a command queue in the system memory;

means for loading a command block into the command queue using the host processor, the command block comprising an input data field that contains input data;

means for performing an operation based on the input data using the adjunct processor to generate a result; and

means for storing the result in the input data field such that at least a portion of the input data is overwritten.

50. (Original) A data processing system as recited in Claim 49, wherein the data processing system comprises a cryptographic data processing system, the adjunct processor integrated circuit comprises a cryptographic processor integrated circuit, and the means for performing the operation based on the input data comprises:

means for performing a hash operation based on the input data using the cryptographic processor to generate a hash value.

51. (Original) A data processing system as recited in Claim 50, wherein the means for storing the result in the input data field comprises:

means for storing the hash value in the input data field such that the at least a portion of the input data is overwritten.

52. (Original) A data processing system as recited in Claim 50, wherein the command block further comprises an input pointer field that contains an address in the system memory of an incoming packet and wherein the means for performing the hash operation comprises:

means for performing the hash operation based on the input data and the incoming packet using the cryptographic processor to generate the hash value.

53. (Original) A data processing system as recited in Claim 52, wherein the command block further comprises an output pointer field that contains an address in the system memory for storing a decrypted packet, the data processing system further comprising:

means for decrypting the incoming packet using the cryptographic processor to generate the decrypted packet;

means for attaching the hash value to the decrypted packet; and

means for storing the decrypted packet with the attached hash value at the address in the system memory contained in the output pointer field.

54. (Original) A cryptographic data processing system that comprises a host processor, a system memory coupled to the host processor, and a cryptographic processor integrated circuit that comprises a local memory and is coupled to the host processor and the system memory, the system further comprising:

means for providing a command queue in the system memory;

means for providing a read address for the command queue and a write address for the command queue;

means for loading a random number sample into the command queue using the cryptographic processor beginning at the write address; and

means for reading the random number sample using the host processor beginning at the read address.

55. (Original) A cryptographic data processing system as recited in Claim 54, wherein the means for loading the random number sample into the command queue using the cryptographic processor beginning at the write address comprises:

means for determining if the write address plus an amount corresponding to a size of a single random number sample equals the read address; and

means for loading the random number sample into the command queue using the cryptographic processor beginning at the write address if the write address plus the amount corresponding to the size of the single random number sample does not equal the read address.

56. (Original) A cryptographic data processing system as recited in Claim 55, further comprising:

means for incrementing the write address by the amount corresponding to the size of a single random number sample using the cryptographic processor if the write address plus the amount corresponding to the size of the single random number sample does not equal the read address, the means for incrementing being responsive to the means for loading the random number sample into the command queue using the cryptographic processor beginning at the write address.

57. (Original) A cryptographic data processing system as recited in Claim 54, wherein the means for reading the random number sample using the host processor beginning at the read address comprises:

means for determining whether the read address is equal to the write address; and

means for reading the random number sample using the host processor beginning at the read address if the read address is not equal to the write address.

58. (Original) A cryptographic data processing system as recited in Claim 57, further comprising:

means for incrementing the read address by an amount corresponding to a size of a single random number sample using the host processor, the means for incrementing being responsive to the means for reading the random number sample using the host processor beginning at the read address.

59 - 60. (Canceled)

61. (Currently Amended) A computer program product for operating a cryptographic data processing system that comprises a host processor, a system memory coupled to the host processor, and a cryptographic processor integrated circuit that comprises a local memory and is coupled to the host processor and the system memory, the computer program product comprising:

a computer readable program medium having computer readable program code embodied therein, the computer readable program code comprising:

computer readable program code for providing a command queue in the system memory;

computer readable program code for loading a command block into the command queue using the host processor;

computer readable program code for executing the command block using the cryptographic processor; and computer readable program code for notifying the host processor that the command block has been executed, the computer readable program code for notifying comprising computer readable program code for updating a completion field in the command block using the cryptographic processor.

62. (Original) A computer program product as recited in Claim 61, further comprising:

computer readable program code for providing a read address for the command queue and a write address for the command queue;

wherein the computer readable program code for loading the command block into the command queue using the host processor comprises computer readable program code for loading the command block into the command queue using the host processor beginning at the write address, and wherein the computer readable program code for executing the command block using the cryptographic processor comprises computer readable program code for executing the command block using the cryptographic processor beginning at the read address.

63. (Original) A computer program product as recited in Claim 62, wherein the computer readable program code for loading the command block into the command queue using the host processor beginning at the write address comprises:

computer readable program code for determining if the write address plus an amount corresponding to a size of a single command block equals the read address; and

computer readable program code for loading the command block into the command queue using the host processor beginning at the write address if the write address plus the amount corresponding to the size of the single command block does not equal the read address.

64. (Original) A computer program product as recited in Claim 63, further comprising:

computer readable program code for incrementing the write address by the amount corresponding to the size of a single command block using the host processor if the write address plus the amount corresponding to the size of the single command block does not equal the read address, the computer readable program code for incrementing being responsive to the computer readable program code for loading the command block into the command queue using the host processor beginning at the write address.

65. (Original) A computer program product as recited in Claim 62, wherein the computer readable program code for executing the command block using the cryptographic processor beginning at the read address comprises:

computer readable program code for determining whether the read address is equal to the write address; and

computer readable program code for executing the command block using the cryptographic processor beginning at the read address if the read address is not equal to the write address.

66. (Original) A computer program product as recited in Claim 65, further comprising:

computer readable program code for incrementing the read address by an amount corresponding to a size of a single command block using the cryptographic processor, the computer readable program code for incrementing being responsive to the computer readable program code for executing the command block using the cryptographic processor beginning at the read address.

67. (Original) A computer program product as recited in Claim 61, wherein the computer readable program code for notifying the host processor that the command block has been executed comprises computer readable program code for invoking an interrupt using the cryptographic processor after executing the command block.

68. (Canceled)

69. (Currently amended) A computer program product as recited in Claim ~~68~~ 61, further comprising:

computer readable program code for providing a periodic interrupt; and computer readable program code for reading the completion field using the host processor upon invocation of the periodic interrupt.

70. (Original) A method as recited in Claim 61, wherein the computer readable program code for notifying the host processor that the command block has been executed comprises:

computer readable program code for setting a timer after loading the command block into the command queue using the host processor; and

computer readable program code for checking whether the command block has been executed after expiration of the timer.

71. (Original) A computer program product as recited in Claim 61, further comprising:

computer readable program code for loading at least one operand from the command queue to the local memory;

computer readable program code for performing at least one operation on the at least one operand to generate a result in the local memory; and

computer readable program code for storing the result generated in the local memory in the command queue.

72. (Original) A computer program product for operating a cryptographic data processing system that comprises a host processor, a system memory coupled to the host processor, and a cryptographic processor integrated circuit that is coupled to the host processor and the system memory, the computer program product comprising:

a computer readable program medium having computer readable program code embodied therein, the computer readable program code comprising:

computer readable program code for providing a command queue in the system memory;

computer readable program code for loading a command block into the command queue using the host processor;

computer readable program code for setting a value of an interrupt field in the command block to request an interrupt when the command block has been executed;

computer readable program code for executing the command block using the cryptographic processor; and

computer readable program code for invoking an interrupt using the cryptographic processor after executing the command block if the interrupt field in the command block is set to the value to request the interrupt.

73. (Original) A computer program product as recited in Claim 72, further comprising:

computer readable program code for storing error information in the command block that is associated with executing the command block using the cryptographic processor.

74 - 76. (Canceled)

77. (Original) A computer program product for operating a data processing system that comprises a host processor, a system memory coupled to the host processor, and an adjunct processor integrated circuit that is coupled to the host processor and the system memory, the computer program product comprising:

a computer readable program medium having computer readable program code embodied therein, the computer readable program code comprising:

computer readable program code for providing a command queue in the system memory;

computer readable program code for loading a command block into the command queue using the host processor, the command block comprising an input data field that contains input data;

computer readable program code for performing an operation based on the input data using the adjunct processor to generate a result; and

computer readable program code for storing the result in the input data field such that at least a portion of the input data is overwritten.

78. (Original) A computer program product as recited in Claim 77, wherein the data processing system comprises a cryptographic data processing system, the adjunct processor integrated circuit comprises a cryptographic processor integrated circuit, and the

computer readable program code for performing the operation based on the input data comprises:

computer readable program code for performing a hash operation based on the input data using the cryptographic processor to generate a hash value.

79. (Original) A computer program product as recited in Claim 78, wherein the computer readable program code for storing the result in the input data field comprises:

computer readable program code for storing the hash value in the input data field such that the at least a portion of the input data is overwritten.

80. (Original) A computer program product as recited in Claim 78, wherein the command block further comprises an input pointer field that contains an address in the system memory of an incoming packet and wherein the computer readable program code for performing the hash operation comprises:

computer readable program code for performing the hash operation based on the input data and the incoming packet using the cryptographic processor to generate the hash value.

81. (Original) A computer program product as recited in Claim 80, wherein the command block further comprises an output pointer field that contains an address in the system memory for storing a decrypted packet, the computer program product further comprising:

computer readable program code for decrypting the incoming packet using the cryptographic processor to generate the decrypted packet;

computer readable program code for attaching the hash value to the decrypted packet; and

computer readable program code for storing the decrypted packet with the attached hash value at the address in the system memory contained in the output pointer field.

82. (Original) A computer program product for operating cryptographic data processing system that comprises a host processor, a system memory coupled to the host processor, and a cryptographic processor integrated circuit that comprises a local memory

and is coupled to the host processor and the system memory, the computer program product comprising:

a computer readable program medium having computer readable program code embodied therein, the computer readable program code comprising:

computer readable program code for providing a command queue in the system memory;

computer readable program code for providing a read address for the command queue and a write address for the command queue;

computer readable program code for loading a random number sample into the command queue using the cryptographic processor beginning at the write address; and

computer readable program code for reading the random number sample using the host processor beginning at the read address.

83. (Original) A computer program product as recited in Claim 82, wherein the computer readable program code for loading the random number sample into the command queue using the cryptographic processor beginning at the write address comprises:

computer readable program code for determining if the write address plus an amount corresponding to a size of a single random number sample equals the read address; and

computer readable program code for loading the random number sample into the command queue using the cryptographic processor beginning at the write address if the write address plus the amount corresponding to the size of the single random number sample does not equal the read address.

84. (Original) A computer program product as recited in Claim 83, further comprising:

computer readable program code for incrementing the write address by the amount corresponding to the size of a single random number sample using the cryptographic processor if the write address plus the amount corresponding to the size of the single random number sample does not equal the read address, the computer readable program code for incrementing being responsive to the computer readable program code for loading the random number sample into the command queue using the cryptographic processor beginning at the write address.

85. (Original) A computer program product as recited in Claim 82, wherein the computer readable program code for reading the random number sample using the host processor beginning at the read address comprises:

computer readable program code for determining whether the read address is equal to the write address; and

computer readable program code for reading the random number sample using the host processor beginning at the read address if the read address is not equal to the write address.

86. (Original) A computer program product as recited in Claim 85, further comprising:

computer readable program code for incrementing the read address by an amount corresponding to a size of a single random number sample using the host processor, the computer readable program code for incrementing being responsive to the computer readable program code for reading the random number sample using the host processor beginning at the read address.